

TITLE OF THE INVENTION  
PRINT CONTROL APPARATUS AND METHOD

FIELD OF THE INVENTION

- 5           The present invention relates to a print control apparatus and method for generating print data in correspondence with, e.g., a drawing command from an application.

10 BACKGROUND OF THE INVENTION

- In a conventional print processing system including a printer capable of designating a plurality of print conditions, e.g., print conditions such as color tone and resolution, these print conditions are
- 15   fixed or explicitly designated by the user. Edit print setting of printing data of two pages on one page at once, or adding a date or predetermined frame to an image and printing them is also explicitly designated by the user.
- 20           As a print processing system of causing a printer having a plurality of print modes, e.g., PDL (Page Description Language) mode and image mode to print data, there is proposed a system having an automatic print mode selection function of automatically selecting a
- 25   print mode used for printing in print processing (specifically, processing by a printer driver) (Japanese Patent Laid-Open No. 11-129583). In the PDL

mode, a host converts application data into PDL data  
containing a command in a format interpretable by a  
printer and transmits the PDL data to the printer. In  
the image mode, the host converts application data into  
5 bitmap data and transmits the bitmap data to the  
printer.

To decide print conditions, the print processing  
system having this automatic print mode selection  
function checks print contents to analyze the type of  
10 document to be printed, and automatically sets a print  
mode based on the analysis result.

This system analyzes print contents and  
determines a print mode even when a print mode which  
should be set in automatic setting due to print  
15 conditions explicitly designated by the user. As a  
result, the time taken to decide a print mode becomes  
long, and the total print time also becomes long  
because of the time for analyzing print contents and  
deciding a print mode.

As for a processing print function which cannot  
be executed in a given print mode as an automatic  
setting choice, the user cannot select a print mode for  
realizing this function when a print mode is  
automatically selected. In this case, no choice is  
25 displayed for the user.

#### SUMMARY OF THE INVENTION

The present invention has been made in consideration of the above situation, and has as its object to provide a print control apparatus and method capable of automatically setting a print mode within a short time, and in automatic setting, preferentially selecting a print mode which satisfies a function and condition designated by the user and performing print processing in the print mode.

To achieve the above object, the present invention has the following arrangement.

According to the present invention, the foregoing object is attained by providing a print control apparatus for generating print data for a print device having a plurality of print modes, comprising: user interface means for causing a user to set a print setting item; first decision means for deciding one of the plurality of print modes in accordance with an item set via the user interface means; and generation means for generating print data to be printed by the print device in the print mode decided by the first decision means.

In a preferred embodiment, The apparatus further comprises transmission means for transmitting the print data generated by the generation means to the print device.

In a preferred embodiment, the apparatus further comprises second decision means for, when the first

decision means does not decide one print mode, deciding  
a print mode in accordance with contents of print data  
to be printed, and when the second decision means  
decides the print mode, the transmission means  
5 transmits the print data to the print device in the  
print mode.

In a preferred embodiment, prior to decision of a  
print mode by the first and second decision means, the  
user interface means causes the user to select one of  
10 the plurality of print modes or an automatic mode in  
which one of the plurality of print modes is  
automatically decided, and the first and second  
decision means decide one print mode when the automatic  
mode is selected.

15 In a preferred embodiment, the item set via the  
user interface means includes an adjustment item for  
adjusting quality of an image to be printed.

In a preferred embodiment, the item set via the  
user interface means includes an edit item for editing  
20 an image to be printed.

In a preferred embodiment, the plurality of print  
modes include an image mode in which the print device  
receives and prints image data, and when the adjustment  
item is set via the user interface means, the first  
25 decision means decides the image mode as a print mode.

In a preferred embodiment, the plurality of print  
modes include a PDL mode in which the print device

receives a page description instruction, generates an image, and prints the image, and when the edit item is set via the user interface means, the first decision means decides the PDL mode as a print mode.

5 In a preferred embodiment, the second decision means calculates data amounts of print data to be transmitted to the print device in the respective image and PDL modes, and decides a mode exhibiting a small data amount as a print mode.

10 Other features and advantages of the present invention will be apparent from the following description taken in conjunction with the accompanying drawings, in which like reference characters designate the same or similar parts throughout the figures  
15 thereof.

#### BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and constitute a part of the specification,  
20 illustrate embodiments of the invention and, together with the description, serve to explain the principles of the invention.

Fig. 1 is a block diagram showing the basic arrangement of a host computer to which a print data  
25 processing apparatus according to an embodiment of the present invention can be applied;

Fig. 2 is a block diagram for explaining the flow

of print data in the print data processing apparatus;

Fig. 3 is a block diagram for explaining the flow of print data in the print data processing apparatus;

Figs. 4A and 4B are schematic views showing  
5 time-series comparison between the behavior of a driver in the arrangement of Fig. 2 and that of a driver in the arrangement of Fig. 3;

Fig. 5 is a view showing an example of the memory map of a RAM in the host computer shown in Fig. 1;

10 Fig. 6 is a view showing an example of a print sample susceptible to an operation mode setting error;

Fig. 7 is a flow chart showing an example of print data processing procedures in a print control apparatus according to the present invention;

15 Fig. 8 is a table showing the display items of a user interface according to the present invention;

Fig. 9 is a view showing an example of a user interface window for a user inquiry according to the embodiment;

20 Fig. 10 is a view showing an example of determination information to be collected in the embodiment;

Fig. 11 is a flow chart showing an example of print data processing procedures in the print control  
25 apparatus according to the present invention;

Fig. 12 is a flow chart showing the example of print data processing procedures in the print control

apparatus according to the present invention;

Fig. 13 is a flow chart showing the example of print data processing procedures in the print control apparatus according to the present invention;

5 Fig. 14 is a flow chart showing the example of print data processing procedures in the print control apparatus according to the present invention;

Fig. 15 is a view showing the memory map of a storage medium storing various data processing programs  
10 which can be read out by a print system to which the print control apparatus according to the present invention can be applied;

Fig. 16 is a flow chart showing an example of print data processing procedures in the print control  
15 apparatus according to the present invention; and

Fig. 17 is a table showing the determination item and operation mode of a user interface according to the embodiment.

## 20 DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

An embodiment to which the present invention can be preferably applied will be described.

Fig. 1 is a block diagram for explaining the arrangement of a host computer including a print data  
25 processing apparatus to which the embodiment of the present invention can be applied. In Fig. 1, a CPU 1 processes a document containing graphics, images,

characters, tables (including a spreadsheet), and the like on the basis of a document processing program or the like stored in the program area of a ROM 3 or an external memory 11. The CPU 1 integrally controls  
5 devices connected to a system bus 4.

The ROM 3 or external memory 11 stores an operating system (to be referred to as an OS hereinafter) serving as a control program for the CPU 1, font data used to process a document, and various other  
10 data. A RAM 2 functions as, e.g., the main memory and work area of the CPU 1. A keyboard controller (KBC) 5 controls an input from a keyboard (KB) 9 or a pointing device such as a mouse (not shown). A CRT controller (CRTC) 6 controls display of a CRT display 10. The  
15 external memory 11 is a hard disk (HD), floppy disk (FD), or the like, and stores a boot program, various applications, font data, user files, edit files, print data generation processing program (to be referred to as a printer driver hereinafter), and the like. A disk  
20 controller (DKC) 7 controls access to the external memory 11 from the CPU 1 or the like. A printer controller (PRTC) 8 is connected to a printer 100 via a predetermined bidirectional interface (interface) 21, and executes communication control processing. The CPU  
25 1 executes, e.g., rasterizing processing of an outline font to a display information RAM set in the RAM 2, and enables WYSIWYG on the CRT 10. Further, the CPU 1



opens various registered windows on the basis of commands designated with a mouse cursor (not shown) on the CRT display 10, and executes various data processes. In executing printing, the user designates a window  
5 (user interface) to be opened that is provided by a printer driver and concerns print setting, sets a printer on the window, and also sets a print processing method including selection of a print mode for the printer driver.

10 Figs. 2 and 3 are block diagrams for explaining the flow of print data in the arrangement of the print data processing apparatus shown in Fig. 1. Figs. 2 and 3 show general print data generation processing in a host computer or dedicated print server connected to a  
15 print device such as a printer directly or via a network.

In Fig. 2, an application 201', graphic engine 202', printer driver 203', and system spooler 204' are program modules which exist as files saved in the  
20 external memory 11 such as a CD-ROM or hard disk, and are loaded to the RAM 2 and executed by an OS or a module using these modules.

The application 201' and printer driver 203' can be added to an FD, a CD-ROM, or the external memory 11  
25 such as a hard disk via a network (not shown). The application 201' saved in the external memory 11 is loaded to the RAM 2 and executed. In printing data

from the application 201' by the printer 100, the data is output (drawn) by using the graphic engine 202' which has been loaded to the RAM 2 and is in an executable state.

5           The graphic engine 202' similarly loads the printer driver 203' prepared for each print device from the external memory to the RAM 2, and converts an output from the application 201' into a printer control command by using the printer driver 203'. The  
10 converted printer control command passes through the system spooler 204' loaded to the RAM 2 by the OS, and is output to the printer 100 via the interface 21.

          A print data processing apparatus shown in Fig. 3 is extended from the print data processing apparatus  
15 shown in Fig. 2. The apparatus of Fig. 3 temporarily generates a spool file 303 made up of intermediate codes when transmitting a print instruction from a graphic engine 202 to a printer driver 203.

          More specifically, in the print data processing  
20 apparatus shown in Fig. 2, the application 201' is released from print processing after the printer driver 203' converts all print instructions from the graphic engine 202' into printer control commands. In the print data processing apparatus shown in Fig. 3, an  
25 application 201 is released from print processing after a spooler 302 converts all print instructions into intermediate code data and outputs them to the spool

file 303. In general, the time up to release of an application (RTA: Return To Application) is shorter in the latter case. However, the time up to the completion of output of a print result from a print device is longer in the latter case by a spool file generation time. These relationships are shown in Fig. 4. In Fig. 4, the behavior of a driver in the arrangement of Fig. 2 and that of a driver in the arrangement of Fig. 3 are schematically compared in time series.

In Fig. 4, the application 201' of Fig. 2 is released from print processing at a timing 401, and the application 201 of Fig. 3 is released from print processing at a timing 411. The difference between these timings is an RTA difference. The end timing of generating print data to be transferred to the print device is the timing 401 in the system of Fig. 2, and in the system of Fig. 3, is timing 412 at which the printer driver 203 completes data generation. The difference between the timings 401 and 412 is a difference in timing of starting printing the first page. If the two systems use the same printer engine, the timing difference directly becomes an output completion timing difference because the time taken for recording operation is equal between the two systems.

The print data processing apparatus shown in Fig. 3 can analyze drawing information within a page

and process data of a generated spool file 303 during generation of an intermediate code to be stored in the spool file 303. This print data processing apparatus enables automatically selecting an optimal operation  
5 mode for print data sent from the application 201 in accordance with function setting performed on a user interface 205. Furthermore, the print data processing apparatus can realize an additional print function not provided to a general application, such as scaling or  
10 N-up printing of reducing a plurality of pages into one page and printing it.

For this purpose, the system of the print data processing apparatus shown in Fig. 2 is so extended as to spool data by intermediate code data, like the print  
15 data processing apparatus shown in Fig. 3. To automatically select an optimal operation mode or process print data, the user performs setting on the window of the user interface 205 provided by the general printer driver 203, the printer driver 203  
20 saves the setting contents in the RAM 2 or external memory 11, and a spool file manager 304 or despooler 305 refers to the values to realize selection of an operation mode or data processing.

The arrangement of Fig. 3 will be explained in  
25 more detail.

In the print data processing apparatus shown in Fig. 3, a dispatcher 301 receives a print instruction

from the graphic engine 202. When the print instruction received by the dispatcher 301 from the graphic engine 202 is a print instruction issued from the application 201 to the graphic engine 202, the dispatcher 301 loads the spooler 302 stored in the external memory 11 to the RAM, and sends the print instruction not to the printer driver 203 but to the spooler 302. The spooler 302 converts the received print instruction (print data) into an intermediate code and outputs the intermediate code to the spool file 303. The spool file 303 is made up of not one file but several files. In converting the print instruction into an intermediate code, the spooler 302 analyzes the converted intermediate code to obtain information such as the type of print instruction, a method of processing the print instruction on the printer side, and the load of this processing. This analysis processing is done for automatic selection of an optimal operation mode. The spooler 302 saves the analysis processing result as another file in the spool file 303. The spooler 302 acquires from the printer driver 203 processing setting information concerning print data set for the printer driver 203, and saves the information in the spool file 303. Note that the spool file 303 is generated as a file in the external memory 11, but can be generated in the RAM 2. The spooler 302 loads the spool file manager 304 stored in

00877003.051001

the external memory 11 to the RAM 2, and notifies the  
spool file manager 304 of the generation status of the  
spool file 303. When the spool file manager 304  
determines that printing is possible by using the  
5 graphic engine 202 again, in accordance with the  
contents of the processing setting information  
concerning print data that is saved in the spool file  
303, the spool file manager 304 loads the despooler 305  
stored in the external memory 11 to the RAM 2, and  
10 instructs the despooler 305 to execute print processing  
of the intermediate code described in the spool file  
303.

The despooler 305 processes the intermediate code  
included in the spool file 303 in accordance with  
15 information of the print instruction analysis result  
included in the spool file 303 and the contents of the  
processing setting information, and maps the  
intermediate code into a print image at a high speed by  
a renderer 306 included in the despooler 305 in  
20 accordance with the print mode (to be described later)  
or the like. Whether to map the intermediate code into  
a print image is comprehensively determined based on  
information such as the set edit setting information,  
and the result file of analysis processing performed by  
25 the spooler 302 that is read out by an optimization  
determination processor included in the despooler 305.  
This determination may be done for each page or each

-

drawing object. In any case, the despooler 305 reads out an intermediate code from the spool file 303, converts it into a form conforming to the API (Application Programming Interface) of the graphic engine 202, and outputs the converted data via the graphic engine 202.

When the print instruction received by the dispatcher 301 from the graphic engine 202 is a print instruction issued from the despooler 305 to the graphic engine 202, the dispatcher 301 switches processing not to the spooler 302 but to the printer driver 203. The printer driver 203 generates a printer control command and outputs it to the printer 100 via the system spooler 204.

Fig. 5 is a view showing an example of the memory map of the RAM 2 in a host computer 200 shown in Fig. 1. Fig. 5 shows a memory map in which a print-related module 504 including a series of print data generation processing programs in this embodiment, an application 501, related data 503, an OS 505, and a BIOS 506 are loaded to the RAM 2 in the host computer 200 and become executable. Reference numeral 502 denotes a free space.

<Decision Processing Based on Setting Items of Print Mode>

Automatic selection processing of an optimal operation mode will be explained. The following automatic selection processing of an operation mode is

implemented by the print processing system in Fig. 3.

In print processing by application software, intermediate print information (intermediate data) independent of a specific PDL is generated from a print instruction and temporarily stored in the spool file 303. The stored intermediate data is analyzed, and an optimal operation mode is automatically decided from the result. As print modes, this embodiment adopts an image mode in which the host performs up to rendering processing of a print image, and a PDL mode in which the print device renders a print image by using a PDL (Printer Description Language) installed in the print device. This printer driver executes automatic selection processing of an optimal operation mode as follows.

Printing of a photograph-containing document by a color LBP will be exemplified. A photographic portion (within the broken line) in Fig. 6 has image data with very high resolution and high tone level. More specifically, this image is an image of 600 dpi and 8 bits for R, G, and B colors each at an equal magnification. The remaining portion has characters. A document as shown in Fig. 6 looks simple at a glance and is considered to be preferably processed in the PDL mode. However, if this image data is converted into a PDL command, image information of an image quality exceeding an image quality printable by a printer, i.e.,



image information having redundant information is sent from the printer driver to the printer. For this reason, such a document can be processed at a higher speed in the image mode. Which of the print modes is better can be determined by obtaining the image data amount sent to the printer in advance.

The spooler of Fig. 3 described in this embodiment generates intermediate data independent of a specific PDL in print processing by application software. At this time, the spooler also spools, from the intermediate data including image data, determination information for predicting two values, i.e., a data amount required when the image data is converted into a PDL command and a data amount required when the image data is mapped by the renderer of the host into a print image in accordance with the output resolution and output tone level of the printer. The despooler 305 predicts and compares two sizes after mapping on the basis of the determination information. The print image is drawn in the print mode exhibiting a smaller value, allowing high-speed print processing. Hence, in processing print data including a high-quality image as shown in Fig. 6, the printer driver of this embodiment automatically selects the image mode in which the host maps an image as an optimal operation mode.

Automatic selection processing of an optimal

operation mode is done in this manner. Automatic selection processing is different from print processing and requires its own time other than the print time. A function which cannot be set in either operation mode must be set not to select the function setting in automatic selection processing.

For this reason, this embodiment causes the user to designate a print processing method including whether to automatically determine a print mode on a UI (User Interface) for designating a print mode and print conditions. This can shorten the print time for a setting when an operation mode need not be automatically set. Also for a setting which cannot be satisfied in either operation mode, a message to this effect can be displayed on the UI. Proper print processing can be provided without degrading performance or performing cumbersome user operation.

The user interface 205 and print mode decision processing implemented by the print system of the embodiment will be described with reference to Fig. 7. The printer driver 203 provides the user interface 205 for performing print setting in accordance with a request from the user or an application. If user interface display processing starts via an OS (Operating System), the printer driver 203 determines whether print setting set via the current user interface 205 is an "automatic selection mode" (S701).

-

The automatic selection mode is for automatically determining and deciding a print mode. When the automatic selection mode is set, actual print processing is done in either of the PDL and image modes.

5           In accordance with the determination result in step S701 and a table (Fig. 8) which describes print conditions settable in each mode, the printer driver 203 provides the user interface 205 for causing the user to set print conditions, and displays the user interface 205 via the OS (S702, S706). In Fig. 8, an item settable by the user in a selected print mode is checked with a circle. For example, if the automatic selection mode or PDL mode is selected, the item "Mode Details" is not displayed as a choice.

10           The printer driver 203 causes the user to set necessary items in steps S703 and S707. In the automatic selection mode, the printer driver 203 decides a print processing method on the basis of the print setting set via the user interface 205 (S704).  
15           Then, the printer driver 203 notifies the dispatcher 301 of the decided choice or a choice selected by the user (S705). At this time, the display on the UI is kept unchanged. Note that the notified choice includes a print mode such as the PDL mode, image mode, or  
20           automatic selection mode. The print processing method is decided in step S704 by the printer driver 203 on the basis of only the print setting set via the user  
25           the basis of only the print setting set via the user

interface 205. In this stage, the print mode may or may not be decided to the PDL or image mode. If the print mode cannot be decided, it keeps the automatic selection mode, and is decided by analyzing

5 intermediate data in spooler processing, as will be described later with reference to Fig. 13.

Details of processing in determination step S704 will be described with reference to Figs. 16 and 17. The table of Fig. 17 shows a print condition including  
10 a user-settable item and value, and a print mode which satisfies the condition. For example, when the value of the item "Mode Details" is set to "High Speed" or "High Quality", the print mode which satisfies this condition is the image mode "Raster", as shown in  
15 Fig. 17. When the item "Overlay Print" is set, the print mode must be the PDL mode. In this fashion, a setting item and set value which restrict a print mode, and a print mode corresponding to the setting are prepared in advance as a table. The setting items  
20 registered in the table of Fig. 17 will be called determination items because these items are determination targets in the procedures of Fig. 16. In this embodiment, the table of Fig. 17 is created to select the image mode when adjustment of the quality of  
25 an image to be printed is designated, and the PDL mode when editing of an image is designated.

Fig. 16 is a flow chart for explaining in detail



Since a print mode suitable for print setting on the user interface 205 provided by the printer driver 203 is easily decided, the processing speed can increase without performing automatic determination by 5 spool processing. Accordingly, the throughput of print processing can increase.

00077003 051109

An example of the UI in this embodiment will be explained with reference to Fig. 9. Even when two 10 print modes 901, CPDL (PDL mode) and Raster (image mode), are provided, an "AUTO" mode which means an automatic setting mode is provided, and the AUTO mode is set as an initial value. As for print conditions displayed at this time, an automatically set item is displayed in a state in which the user cannot select 15 any value, as represented by a box 902, and an item explicitly settable by the user is displayed in a selectable state, as represented by a box 903.

A choice 904 which can be processed in only the Raster (image) mode is also displayed in a selectable 20 state. If the user clicks "OK" in this state, step S704 of Fig. 7, i.e., the procedures of deciding a print processing method in Fig. 16 are executed because the print mode 901 is set to "Auto". In Fig. 9, the item "Color Halftone" is set to "Error Diffusion". 25 When the item "Color Halftone" is set to "Error Diffusion", only the image mode "Raster" is selected as a print mode from the table of Fig. 17. Thus, the

print mode is decided to the image mode. The dispatcher 301 is notified of this decision.

When no print mode is automatically decided, print mode decision processing in despool processing is performed, as will be described with reference to Fig. 13. When a print mode is decided in step S704 of Fig. 7, the dispatcher 301 causes the printer driver 203 to start processing in accordance with the print mode notified in step S705. That is, if the print mode is the automatic selection processing, the dispatcher 301 transfers a print instruction to the spooler 302 and causes the spooler 302 to perform spool processing. If the print operation mode is the "PDL mode" or "image mode", the dispatcher 301 transfers a print instruction to the printer driver 203 and causes the printer driver 203 to perform print data generation processing.

#### <Determination Information>

If an operation mode is automatically decided, i.e., if no print mode is decided in step S704 of Fig. 7 and the print mode notified in step S705 is the "automatic selection mode", the spooler 302 collects determination information described above. In generating intermediate data of each page, the spooler 302 collects determination information as shown in Fig. 10 that is necessary for deciding an optimal operation mode, and records the collected information together with the intermediate data. The determination





maximum point size of a character with a certain threshold, when the maximum point size is larger than the threshold, determines that rendering processing on the host side, i.e., the image mode is efficient, and

5 when the maximum point size is smaller, determines that rendering processing on the printer side, i.e., the PDL mode is efficient. As for graphics, the despooler 305 compares the number of objects with a certain threshold, when the number of objects is larger than the threshold,

10 determines that rendering processing on the host side, i.e., the image mode is efficient, and when the number of objects is smaller, determines that rendering processing on the printer side, i.e., the PDL mode is efficient. As for an image, the despooler 305 predicts

15 a data amount to be transmitted to the printer as a PDL command on the basis of information such as the resolution, tone level, and drawing region, when the predicted value is larger than an image size obtained by rendering processing and halftoning processing on

20 the host side, determines that rendering processing on the host side, i.e., the image mode is efficient, and when the predicted value is smaller, determines that rendering processing on the printer side, i.e., the PDL mode is efficient. Note that the printer used is

25 assumed to be a binary printer. As for ROP, if an ROP number not supported by the printer is designated, the despooler 305 determines that rendering processing on

-  
the host side, i.e., the image mode should be done;  
otherwise, determines that rendering processing on the  
printer side, i.e., the PDL mode is efficient. A mode  
determined to be efficient is selected as a print mode.

5 <Print Mode Decision Processing Based on Determination  
Information>

A series of steps of generating and reading out  
intermediate data in the present invention will be  
described with reference to the flow chart of Fig. 11.

10 In step S111 of Fig. 11, print data generation  
processing is performed. In step S112, the printer  
driver inquires a print result of the printer.  
Figs. 12, 13, and 14 are flow charts for explaining  
step S111 of Fig. 11 as print processing in the  
15 arrangement of Fig. 3. The processing flow in Fig. 12  
shows processing by the spooler in Fig. 3, the  
processing flow in Fig. 13 shows processing by the  
despooler in Fig. 3, and the processing flow in Fig. 14  
shows processing by the printer driver in Fig. 3.  
20 <Spool Processing>

In the flow chart of Fig. 12, various  
initialization operations are done at the start of  
processing, and print processing starts. The spooler  
receives print data from the graphic engine, and spools  
25 the contents as intermediate print information (print  
data) independent of a specific PDL (S1201). If the  
print mode is not decided yet and keeps "Auto", the

spooler collects pieces of determination information shown in Fig. 10 (S1202). The spooler executes steps S1201 and S1202 for all data within the page (S1203). Upon the completion of processing of one page, the

5 spooler spools as a file the pieces of determination information collected in step S1202 (S1204). The spooler has spooled information of one page, so that the despooler which operates as another process is activated (S1205). If the despooler has already been

10 activated, no processing is done in step S1205. Since the despooler operates as another process in another time series, processing by the despooler will be explained with reference to Fig. 13. In step S1206, the spooler checks whether all pages have been

15 processed. If NO in step S1206, the spooler repeats processing from step S1201 again; if YES, ends processing. If a print mode has already been decided by the procedures of Fig. 7, the spooler may spool the print mode in addition to the general information 1001

20 concerning a print job in step S1204. In this case, the despooler and printer driver perform processing in the print mode spooled at this time.

<Despool Processing and Rendering Processing>

The processing flow of the despooler will be

25 described with reference to the flow chart of Fig. 13. Processing in Fig. 13 starts when the print mode is not decided in Fig. 7 and is "Auto". If the print mode is

not "Auto", the printer driver generates print data in either the set "PDL mode" or "image mode".

Various initialization operations are done at the start of processing, and despool processing starts.

- 5 The despooler reads out determination information of each page and decides a print mode (S1302). In the automatic selection mode, a decision in step S1302 is made by the method described in <Determination Information>. In a mode other than the automatic selection mode, the set mode is directly selected as a decided print mode.
- 10

- In step S1303, the despooler determines the print mode in accordance with the result. The despooler shifts to step S1304 for a page or job in the PDL mode that should be processed on the printer side (PDL), and to step S1307 for a page or job in the image mode that should be processed on the host side.
- 15

- In step S1304, the despooler opens a file including the spooled contents of drawing data and various drawing attributes, and reads out the contents from the start of the file. The despooler converts the spooled intermediate data into an API form provided by the graphic engine, and performs print processing again by using the function of the graphic engine (S1305).
- 20
- 25 The despooler checks in step S1306 whether print information of one page has been reconstructed, and if necessary, repeats processing in steps S1304 and S1305.

For a page determined to be rendered on the host side as a result of determination for each page, the despooler shifts to step S1307, opens a file including the spooled contents of drawing data and various  
5 drawing attributes, and reads out the contents from the start of the file. The despooler converts the spooled intermediate data into an API form provided by the renderer, and renders each print information so as to obtain a print image (S1308). The rendering result is  
10 an image. The despooler converts this image into an API form provided by the graphic engine, and performs print processing again by using the function of the graphic engine. In step S1310, the despooler checks whether information of one page has been processed. If  
15 YES in step S1310, the despooler checks in step S1311 whether all pages have been processed. If NO in step S1311, the despooler repeats processing from step S1303. Then, the despooler ends processing.  
<Drawing Command Generation Processing>

20 In the processing flow of Fig. 13, drawing processing is done by the API call of the graphic engine. This processing is executed as another process. More specifically, the graphic engine is dynamically linked to the printer driver and executes print  
25 processing. The processing flow of the printer driver will be explained with reference to Fig. 14. Fig. 14 is a flow chart showing an example of print data

processing procedures in the print control apparatus according to the present invention.

- 5 The printer driver 203 receives a drawing instruction from the application 201 via the graphic engine 202 (S1401), and generates a printer control (drawing) command data corresponding to each drawing instruction (e.g., line drawing) (S1402), and causes the system spooler 204 in the RAM 2 or external memory 11 to spool the printer control command data (S1403).
- 10 The printer driver 203 checks whether drawing processing by the application 201 and graphic engine 202 ends (S1404). If NO in step S1404, the printer driver 203 returns to step S1401 and repeats the same processing.
- 15 If YES in step S1404, the printer driver 203 transmits to the printer 100 the printer control command data spooled by the system spooler 204 (S1405), and ends processing. The feature of the printer control command generation method (to be referred to as
- 20 a vector graphics print mode hereinafter) executed by the printer driver 203 is that a printer control (drawing) command which provides a geometrical expression corresponding to vector graphics such as lines is transmitted to the printer 100 and the
- 25 internal drawing processing system and program of the printer 100 generate (rasterize) a print image. In this case, the time taken for print processing in the

printer 100 is decided by the type and number of objects.

In the image mode, the printer driver transmits image data received from the graphic engine, to the  
5 system spooler without converting the image data to PDL.

In this way, the print processing system according to the embodiment can decide a print mode based on print conditions designated by the user even when automatic setting is designated as a print mode.  
10 Cumbersome operation of analyzing spooled data and deciding a print mode can be eliminated depending on set print conditions, and the throughput of the entire print processing can increase. If no print mode can be decided based on print conditions designated by the  
15 user, a print mode is decided by analyzing spooled data. Thus, a print mode suitable for print data can be decided.

The host computer may comprise a plurality of image drivers, a plurality of different types of PDL  
20 drivers, or a mixture of pluralities of image drivers and PDL drivers.

Fig. 15 is a view for explaining the memory map of a storage medium storing various data processing programs which can be read out by a print system to  
25 which the print control apparatus according to the present invention can be applied.

Although not shown, the storage medium may store

information for managing programs stored in the storage medium, such as version information and user information, and information depending on the OS of a program readout side, such as an icon for identifiably  
5 displaying a program.

Data subordinate to various programs are also managed by the directory. When a program for installing various programs into a computer, and a program to be installed are compressed, a decompression  
10 program or the like may be stored.

Processes shown in Figs. 7, 11, 12, 13, and 14 in this embodiment may be achieved by a host computer in accordance with an externally installed program. The present invention is applied even when pieces of  
15 information including a program are supplied to an output device from a storage medium such as a CD-ROM or flash memory or from an external storage medium via a network.

The object of the present invention is also  
20 achieved when a storage medium which stores software program codes for realizing the functions of the above-described embodiment is supplied to a system or apparatus, and the computer (or the CPU or MPU) of the system or apparatus reads out and executes the program  
25 codes stored in the storage medium.

In this case, the program codes read out from the storage medium realize new functions of the present



invention, and the storage medium which stores the program codes constitutes the present invention. As a storage medium for supplying the program codes, a floppy disk, hard disk, optical disk, magnetooptical disk, DVD, CD-ROM, magnetic tape, nonvolatile memory card, ROM, EEPROM, or the like can be used.

The functions of the above-described embodiment are realized not only when the computer executes the readout program codes, but also when the OS (Operating System) running on the computer performs part or all of actual processing on the basis of the instructions of the program codes. The functions of the above-described embodiment are also realized when the program codes read out from the storage medium are written in the memory of a function expansion board inserted into the computer or the memory of a function expansion unit connected to the computer, and the CPU of the function expansion board or function expansion unit performs part or all of actual processing on the basis of the instructions of the program codes.

<Effect of the Invention>

As has been described above, according to the present invention, a print control system capable of automatically setting a print mode and print conditions can automatically set a print mode within a short time. Moreover, this system can preferentially select a print mode which satisfies a function and condition

designated by the user and perform print processing in the print mode.

As many apparently widely different embodiments of the present invention can be made without departing from the spirit and scope thereof, it is to be understood that the invention is not limited to the specific embodiments thereof except as defined in the appended claims.

0877093-061101